

Recommender System using Collaborative Filtering Methods: A Performance Evaluation

Mr. G. Suresh

Assistant professor,

Department of Computer Application,
St. Joseph's College of Arts and Science (Autonomous),
Cuddalore, Tamilnadu.

D. Yogeswary

M.Phil.Scholar,

PG and Research Department of Computer science,
St. Joseph's College of Arts and Science (Autonomous),
Cuddalore, Tamilnadu.

N. Martin Paul Raj

M.Phil.Scholar,

PG and Research Department of Computer science,
St. Joseph's College of Arts and Science (Autonomous),
Cuddalore, Tamilnadu.

Abstract: This paper implements the user-based and item-based collaborative filtering algorithms that make up the Netflix recommender system, and describe its business purpose and also the role of search and related algorithms, which for us turns into a Recommendations problem as well. This paper evaluates the prediction qualities of user-based and item-based collaborating filtering methods using Netflix dataset and gives the recommendations to the active user by taking advantage of past usage experience of subscriber Rating users. The experimental results show that item-based collaborative filtering method is better than user-based approach in terms of better prediction accuracy.

Keywords: Recommender System, Collaborative Filtering, Rating, and Rating recommendation.

1. INTRODUCTION

Internet television is about selection: what to watch, when to watch, and where to watch, compared with linear transmission and cable systems that offer whatever is now playing on perhaps 5 to 10 favorite channels. But humans are unpredictably bad at picking between many possibilities, quickly getting overwhelmed and picking "none of the above" or making poor choices. Meanwhile, an advantage of internet is that it can carry videos from a larger catalog appealing to a wide range of demographics and tastes, and including niche titles of interest only to relatively small groups of users.

With increasing the number of movies and television shows, rating is usually employed for describing non-functional characteristics of movies and television shows. Among different rating properties of movies and television shows, some properties are user self-determining and have same values for different users (e.g. popularity, availability, etc.). The values of the user self-determining rating properties are usually offered by rating providers. On the other hand, some rating properties are user dependent and have dissimilar values for dissimilar users (e.g., response time, invocation failure rate, etc.). Obtaining values of the user needy rating

properties is a challenging mission, since real world Movies and television shows estimation in the client side is regularly required for computing the performance of the user needy rating properties of Movies and television shows. Client-side movies and television shows, the estimation wants the real-world movies and television shows invocations and encounters the following drawbacks:

- Consumer research suggests that a typical Netflix member loses interest after perhaps 60 to 90 seconds of choosing, having reviewed 10 to 20 titles (perhaps 3 in detail) on one or two screens
- The user either finds somewhat of curiosity or the risk of the user abandoning our rating increases substantially

However, without sufficient client-side evaluation, accurate values of the user-dependent rating properties cannot be obtained. Optimal movies and television shows selection and recommendation are complicated to achieve. To attack this critical challenge, we recommend a collaborative filtering based approach for making personalized rating value prediction for the rating users. Collaborative filtering [1] is the method which without human intervention predicts values of the present user by collecting the information from other related users or items.

Well-known collaborative filtering methods consist of user-based approach [2], [3], [4] and item-based approach [5], [6], and [7]. Due to their great successes in modeling uniqueness of users and items, collaborative filtering techniques have been broadly engaged in famous commercial systems, such as Amazon, 1 E-bay, 2 etc.

In this paper, we systematically merge the user-based approach [2], [3], [4] and item-based approach [5], [6], [7]. for predicting the rating values for the present user by employing historical movies and television shows rating data from other related users and related movies and television shows. Similar rating users are defined as the rating users who have similar historical rating knowledge on the same set of frequently invoked movies and television shows with the

present user. Different from the old-style movies and television shows evaluation approaches [7], our approach predicts user needy rating values of the target movies and television shows without requiring real-world Movies and television shows invocations.

The Movies and television shows rating values obtained by our approach can be employed by other rating driven approaches (e.g., Movies and television shows selection fault tolerant Movies and television shows etc.). The part of this paper enclosed three-fold:

- First, we propose a user-collaborative mechanism for collecting historical rating data of Movies and television shows from different rating users.
- Second, we propose a movies and television shows rating value prediction approach by combining the old-style user-based and item-based collaborative filtering methods. Our approach require no movies and televisionshows invocations and can help rating users determinethe suitable movies and television shows by examining rating information from similar users.
- Finally, we conduct a large-scale real-world experimental analysis for verifying our rating prediction results. 100 real-world Movies and television shows s in 22 countries are evaluated by 150 rating users in 24 countries. 1.5 million Movies and television shows invocations are executed by these rating users and detailed experimental results are recounted. To the best of our knowledge, the scale of our experiment is the largest among the published work of movies and television shows rating evaluation and prediction. Our real-world rating data set has been released online for helping futureresearch andmaking our experiments reproducible. The rest of this paper is organized as follows: Section2introduces a user-collaborative rating data collection mechanism. Section3 presents the similarity computation method conclude the paper.

1.1 Related work and discussion

In this section to present some of the research literature related to collaborative filtering, recommender systems, data mining and personalization.

Tapestry [8] is one of the initial implementations of collaborative filtering based on recommender systems. This system trusted on the explicit opinions of people from a close-knit community, such as an office workgroup. However, recommender system for large group of people cannot depend on each person knowing the others. Later, several ratings-based automatic recommender systems were developed. The Group Lens research system [9] provides a Pseudonymous collaborative filtering solution for Usenet news and movies. Ringo [10] and Video Recommender [11] are email and web-based systems that make recommendations on music and movies respectively. A special issue of Communications of the ACM [12] presents a number of different recommender systems.

Other technologies have also been useful to recommender systems, including Bayesian networks, clustering, and Horting. Bayesian networks make a model

based on a training set with a choice tree at each node and edges on behalf of user information. The model can be constructed off-line over a matter of hours or days. The resultant modelis very small, very fast, and fundamentally as accurate as adjacent neighbor methods [13]. Bayesian networks may prove practical for surroundings in which knowledge of user favorites changes slowly with respect to the time needed to build the model but are not appropriate for environments in which user preference models must be updated speedily or often.

Clustering methods work by classifying groups of users who seem to have similar preferences. Once the clusters are formed, predictions for aseparate can be made by averaging the opinions of the other users in that cluster. Some clustering methodscharacterize each user with partial contribution in several clusters. The prediction is then an average across the clusters, weighted by grade of contribution. Clustering methodsregularlyyieldthe less-personal recommendations than other approaches, and in some cases, the clusters have poorer accuracy than nearest neighbor algorithms [14]. Once the clustering is complete, however, performance can be very respectable, since the scope of the group that must be analyzed is much smaller. Clustering techniques can also be applied as a "first step" for 2 decrease the candidate set in a nearest neighbor algorithm or for distributing nearest-neighbor computation across several recommender trains. While in-between the population into clusters may hurt the correctness or recommendations to users near the fringes of their assigned cluster, pre-clustering may be a valuable trade-off among the accuracy and throughput.

Horting is a graph-based method in which nodes are users and edges between nodes specify the degree of parallel between two users [15]. Predictions are produced by mobile the graph to immediate nodes and merging the opinions of the immediate users. Hortingvaries from nearby neighbor as the graph may be walked through other users who have not rated the item in query, thus exploring transitive relationships that nearby neighbor algorithms do not reflect. In one study using artificial data, Horting produced better predictions than a nearest neighbor algorithm [1].

Schafer et al., [19] present a detailed classification and examples of recommender systems used in E-commerce andhow they can deliver one-to-one personalization and at the same can capture the customer faithfulness. Although these systems have been fruitful in the past, their wide use has exposed some of their restrictions such as the problems of sparsity in the data set, problems associated with high dimensionality and so on. Sparsity problem in recommender system has been addressed in [16]. The problems associated with high dimensionality in recommender systems have been debated in [17], and application of dimensionality discountmethods to address these concerns has been examined in [18].

Our work discovers, the experimental results show that user-based collaborative filtering method is better than item-based approach in terms of better prediction accuracy.

1.2 Contributions

This paper has three primary research contributions:

1. Analysis of the user-based prediction algorithms
2. Analysis of the item-based prediction algorithms
3. An experimental comparison of the quality of item-based and user-based finally result shows the better prediction accuracy between them.

2. COLLABORATIVE FILTERING BASED ON RECOMMENDER SYSTEMS

Recommender systems apply data analysis methods to the problem of serving users find the objects they would like to watch at movies and television shows by creating a predicted likeliness score or a list of *top-N* recommended items for a given user. Item recommendations can be made using not the same approaches. Recommendations can be based on demographics of the users, overall top marketing items, or past purchasing custom of users as a predictor of upcoming items. Collaborative Filtering (CF) [10] is the most fruitful recommendation method to date. The basic idea of CF-based algorithms is to deliver item recommendations or predictions based on the opinions of other like-minded users. The opinions of users can be obtained *explicitly* from the users or by using some *implicit* measures.

2.1 Overview of the Collaborative Filtering Process

The aim of a collaborative filtering algorithm is to recommend different items to an active user based on the user's earlier liking and the similar taste of other like-minded users. Opinion can be openly given by the user as rating score or it can be indirectly congenial from the acquisition records by analyzing timing logs, by mining web hyperlinks and so on [9]

3. USER-COLLABORATIVE RATING COLLECTION

To make accurate rating value prediction of Movies and television shows without real-world movies and television shows calls, we need to gather past movies and television shows rating information from other rating users. Though, it is challenging to collect movies and television shows rating material from dissimilar rating users due to: 1) Movies and television shows are scattered over the Internet and are held by dissimilar establishments. 2) Rating users are commonly remote from each other. 3) The current movies and television shows architecture does not offer any mechanism for the movies and television shows rating information sharing. Stimulated by the recent success of YouTube4 and Wikipedia, 5 we propose the concept of user-collaboration for the movies and television shows rating material sharing among rating users. The idea is that, instead of contributing videos (YouTube) or information (Wikipedia), the rating users are encouraged to donate their independently observed past movies and television shows.

Rating data collection mechanism, which are introduced as follows?

1. A Rating user donates past movies and television shows rating data to a centralized server Netflix recommender system [40]. In the following of this paper, the Rating users who require Rating Value prediction Ratings are named as active users.
2. Netflix recommender system [20] picks similar users from the training users for the active user. Training users signify the rating users whose rating values are stored in the Netflix recommender system server and employed for making value predictions for the active users.
3. Netflix recommender system predicts rating values of Movies and television shows for the active user
4. Netflix recommender system makes movies and television shows recommendation based on the predicted rating values of different movies and television shows
5. The Rating user receives the predicted rating values as well as the recommendation results, which can be employed to assist decision making (e.g., Rating selection, composite rating performance prediction, etc.)

In our user-collective mechanism, the active users who donate more movies and television shows rating data will obtain more exact rating value predictions. By this way, the rating users are cheered to contribute their past movies and television shows rating data. More architecture and implementation details of Netflix recommender system will be introduced.

4. SIMILARITY COMPUTATIONS

This section presents the similarity computation method of dissimilar rating users as well as dissimilar Movies and television shows

4.1 Pearson Correlation Coefficient

Given a recommender system consisting of M training users and N movies and television shows items, the relationship between rating users and movies and television shows items is represented by an $M * N$ matrix, called the user-item matrix. Each entry in this matrix $r_{u,i}$ represents a vector of rating values (e.g., response time, failure rate, etc.) that is detected by the rating user upon the movies and television shows item i . If user u did not appeal the movies and television shows item i before, then $r_{u,i} = \text{null}$. In the case that a movies and television shows contains multiple operations, every item (column) of the user-item matrix represents a movies and television shows operation instead of a movies and television shows. Pearson Correlation Coefficient (PCC) has been introduced a number of recommender systems for similarity computation, since it can be effortlessly implemented and can achieve high correctness. In user-based collaborative filtering methods for movies and television shows, PCC is employed to compute the likeness between two rating users and user based on the movies and television shows items they usually invoked using the following equation:

$$Sim(a, u) = \frac{\sum_{i \in I} (r_{a,i} - \bar{r}_a)(r_{u,i} - \bar{r}_u)}{\sqrt{\sum_{i \in I} (r_{a,i} - \bar{r}_a)^2} \sqrt{\sum_{i \in I} (r_{u,i} - \bar{r}_u)^2}}$$

Where $I = I_a \cap I_u$ is the subset of movies and televisionshows items which both user a and user u have invoked beforehand, $r_{a,i}$ is a vector of rating values of movies and televisionshows item i observed by rating user a, \bar{r}_a and \bar{r}_u and r represent average rating values of dissimilar movies and televisionshows observed by rating user a and u, respectively. From this definition, the similarity of two rating users, $Sim(a, u)$ is in the interval of [-1, 1], where a superior PCC value indicates that rating user a, u are more similar. When two rating users have null movies and televisionshows intersection ($I = \text{null}$), the value of $Sim(a, u)$ cannot be strong-minded to ($Sim(a, u) = \text{null}$), since we do not have material for the similarity computation.

Item-based collaborative filtering methods using PCC[5], [7] are similar to the user-based methods. The variance is that item-based methods employ the similarity between the movies and televisionshows items instead of the rating users. The similarity computation of two Movies and televisionshows items i and j can be calculated by

$$Sim(i, j) = \frac{\sum_{u \in U} (r_{u,i} - \bar{r}_i)(r_{u,j} - \bar{r}_j)}{\sqrt{\sum_{u \in U} (r_{u,i} - \bar{r}_i)^2} \sqrt{\sum_{u \in U} (r_{u,j} - \bar{r}_j)^2}}$$

Where $Sim(i, j)$ is the likeness between movies and televisionshows item i and j, $U = (U_i \cap U_j)$ is the subset of rating users who have invoked together by movies and televisionshows item i and movies and televisionshows item j beforehand, and \bar{r}_i signifies the normal rating values of the movies and televisionshows item i detected by dissimilar rating users. $Sim(i, j)$ is also in the interval of [-1, 1]. When two ratings items have null rating user intersection ($U = \text{null}$), the value of $Sim(i, j)$ cannot be calculated ($Sim(i, j) = \text{null}$).

5. EXPERIMENTAL EVALUATION

To evaluate the user-based and item-based in terms of better prediction accuracy using Netflix data set.

Movie data: Using the data from the Netflix recommender system, Netflix is a web-based research recommender system. Every week hundreds of users visit Netflix to rate and receive recommendations for movies. The site now has over 44000 users who have expressed opinions on 3600+ different movies. We randomly selected enough users to obtain 100, 000 ratings from the database (we have only considered users that had rated 20 or more movies). We separated the database into a training set and a test set. For this purpose, to introduce a variable that defines what percentage of data is used as training and test sets, we call this variable x. A value of $x = 0.7$ would indicate 70% of the data was used as training set and 30% of the data was used as test set. The data set was converted into a user-item matrix A

that had 943 rows (i.e., 943 users) and 1682 columns (i.e., 1682 movies that were rated by at least one of the users)

5.1 Performance Comparison

To using Mean Absolute Error (MAE) measure to prediction quality of methods. To study the prediction performance, we compare our two approaches

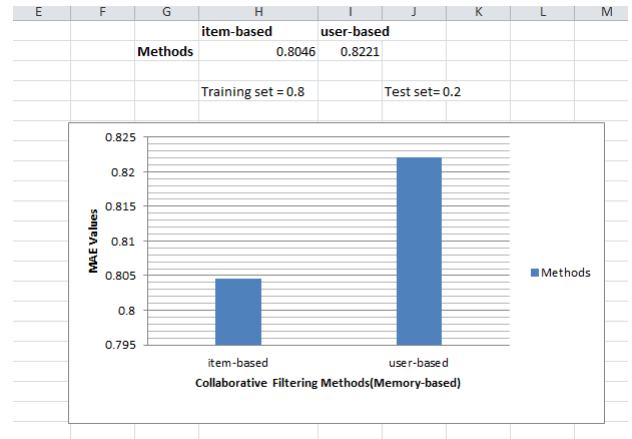


Figure 1 Impact of the training and test set prediction.

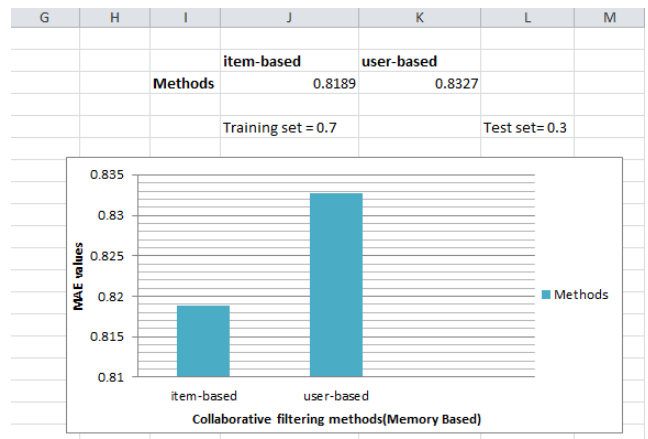


Figure 2 Impact of the training and test set prediction.

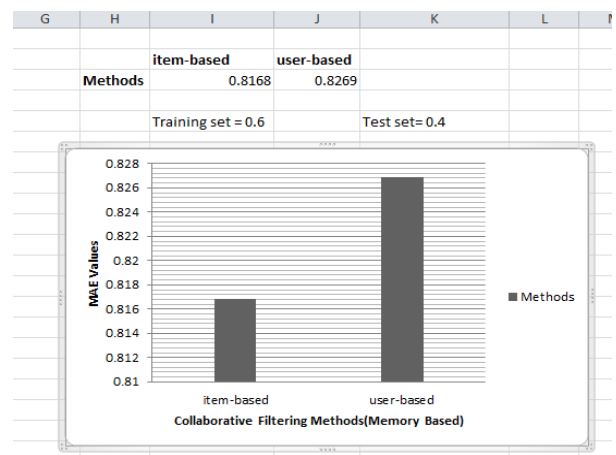


Figure 3 Impact of the training and test set prediction.

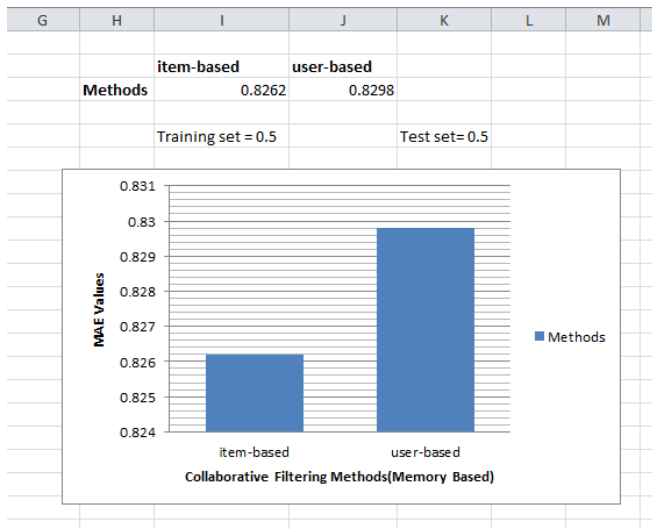


Figure 4 Impact of the training and test set prediction.

6. CONCLUSION

Recommender systems are a powerful new technology for mining additional value for a business from its user databases. These systems help users find movies and television they want to watch. Recommender systems help the users by permitting them to find movies and television shows that they like.

In this paper, we used User-based and Item-Based collaborative methods implemented using Netflix dataset perhaps 60 to 90 seconds of choosing, having reviewed 10 to 20 titles (perhaps 3 in detail) on one or two monitors. The result shows the prediction accuracy with respect to item-based collaborative filtering method is better than user-based approach in terms of better prediction accuracy. In future it is proposed by combining User-based and Item-Based methods also using slope methods.

7. REFERENCE

- [1] J.L. Herlocker, J.A. Konstan, A. Borchers, and J. Riedl, "An Algorithmic Framework for Performing Collaborative Filtering," Proc. 22nd Int'l ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR '99), pp. 230-237, 1999.
- [2] J.S. Breese, D. Heckerman and C. Kadie, "Empirical Analysis of Predictive Algorithms for Collaborative Filtering," Proc. 14th Ann. Conf. Uncertainty in Artificial Intelligence (UAI '98), pp. 43-52, 1998.
- [3] R. Jin, J. Y. Chai, and L. Si, "An Automatic Weighting Scheme for Collaborative Filtering" Proc. 27th Int'l ACM SIGIR Conf.

- Research and Development in Information Retrieval (SIGIR '04), pp. 337-344, 2004.
- [4] G. Xue, C. Lin, Q. Yang, W. Xi, H. Zeng, Y. Yu, and Z. Chen, "Scalable Collaborative Filtering Using Cluster-Based Smoothing," Proc. 28th Int'l ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR '05), pp. 114-121, 2005.
- [5] M. Deshpande and G. Karypis, "Item-Based Top-N Recommendation," ACM Trans. Information System, vol. 22, no. 1, pp. 143-177, 2004.
- [6] G. Linden, B. Smith, and J. York, "Amazon.com Recommendations: Item-to-Item Collaborative Filtering," IEEE Internet Computing, vol. 7, no. 1, pp. 76-80, Jan./Feb. 2003.
- [7] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Item-Based Collaborative Filtering Recommendation Algorithms," Proc. 10th Int'l Conf. World Wide Web (WWW '01), pp. 285-295, 2001.
- [8] Goldberg, D., Nichols, D., Oki, B. M., and Terry, D. (1992). "Using Collaborative Filtering to Weave an Information Tapestry." *Communications of the ACM*, December.
- [9] Konstan, J., Miller, B., Maltz, D., Herlocker, J., Gordon, L., and Riedl, J. (1997). GroupLens: Applying Collaborative Filtering to Usenet News. *Communications of the ACM*, 40(3), pp. 77-87.
- [10] Shardanand, U., and Maes, P. (1995). Social Information Filtering: Algorithms for Automating 'Word of Mouth'. In *Proceedings of CHI '95*. Denver, CO.
- [11] Hill, W., Stead, L., Rosenstein, M., and Furnas, G. (1995). Recommending and Evaluating Choices in a Virtual Community of Use. In *Proceedings of CHI '95*.
- [12] Resnick, P., and Varian, H. R. (1997). Recommender Systems. Special issue of *Communications of the ACM*. 40(3).
- [13] Breese, J. S., Heckerman, D., and Kadie, C. (1998). Empirical Analysis of Predictive Algorithms for Collaborative Filtering. In *Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence*, pp. 43-52.
- [14] Aggarwal, C. C., Wolf, J. L., Wu K., and Yu, P. S. (1999). Horting Hatches an Egg: A New Graph-theoretic Approach to Collaborative Filtering. In *Proceedings of the ACM KDD'99 Conference*. San Diego, CA. pp. 2011-2012.
- [15] Schafer, J. B., Konstan, J., and Riedl, J. (1999). Recommender Systems in E-Commerce. In *Proceedings of ACM E-Commerce 1999 conference*.
- [16] Sarwar, B. M. Konstan, J. A., Borchers, A. Herlocker, J. Miller, B. and Riedl, J. (1998). Using Filtering Agents to Improve Prediction Quality in the GroupLens Research Collaborative Filtering System. In *Proceedings of CSCW '98*, Seattle, WA.
- [17] Billsus, D., and Pazzani, M. J. (1998). "Learning Collaborative Information Filters. In *Proceedings of ICML '98*, pp. 46-53.
- [18] Sarwar, B. M., Karypis, G., Konstan, J. A., and Riedl, J. (2000). "Application of Dimensionality Reduction in Recommender System—A Case Study" In *ACM WebKDD 2000 Workshop*.
- [19] Schafer, J. B., Konstan, J., and Riedl, J. (1999). "Recommender Systems in E-Commerce. In *Proceedings of ACM E-Commerce '1999 conference*.
- [20] C. a Gomez-uribe and N. Hunt, "The Netflix Recommender System: Algorithms, Business Value," vol. 6, no. 4, 2015.